

Atelier N°1(Rappel JS)

On rappelle quelques principes de base de JavaScript :

document.getElementById(String id) : Retourne un objet HTML à partir de son id, défini dans la propriété id de la balise de l'objet. Ne pas confondre la propriété id et la propriété name.

Exemple : saisir et analyser le code qui modifie puis lit la valeur d'un champ à partir de son id.

ATTENTION : ce code contient quelques erreurs de syntaxe, pour les trouver il faut exécuter la page sous Firefox ensuite appuyer sur CTRL + SHIFT + K pour ouvrir le débogueur qui donne les erreurs JavaScript ainsi que la ligne de l'erreur. Voir la capture suivante :

```
<p> <h1> Rappel sur JavaScript</h1> </p>

Entrer un texte : <input type="text" id="nom"><br><br>

<input type="button" onclick="titi()" value="modifier le champ">

<script>

function toto()
{
  var obj = document.getElementById("Non");
  alert('le champ a valeur : '+obj.value);
  obj.value="autre valeur";
  alert('maintenant il contient : '+obj.valu);
}
</script>
```

document.getElementsByTagName(String balise) :

Retourne un **tableau** (Array) d'objets HTML à partir de nom de la balise (tag en anglais). On peut parcourir le tableau retourné avec une boucle et en jouant sur l'indice.

Exemple : saisir et analyser le contenu de code :

Bonjour, ceci est un petit <I>exemple</I> de code !

```
<SCRIPT>
var t=document.getElementsByTagName("span");
alert("Il y a "+t.length+" balises SPAN dans ce document");
for (i=0 ; i<t.length ; i++)
{
  alert("Contenu HTML du span n°" + (i+1) + ":" + t[i].innerHTML);
  alert("Contenu textuel du span n°" + (i+1) + ":" + t[i].textContent);
}
```

```
}  
</SCRIPT>
```

Exercices

Placer le fichier « exemple.html » dans le dossier TP1 et ouvrir la page avec Firefox.
Voici le travail demandé :

Quand on clique sur le bouton « Augmenter la taille du titre (28px) » il faut que le titre de la page (Lorem ipsum !) change de taille afin d'avoir 28px. Pour cela on va modifier sa propriété de style css « **Font-Size** » et mettre '28px'.

Ouvrir le fichier « exemple.html » sous Webstorm. Par quelle balise est entouré le titre ? _____ Quelle est son id ? _____

Créer un nouveau fichier de type « JavaScript », saisir et compléter le code suivant :

```
function augmenter_titre()  
{  
    .....  
    .....  
}
```

Indication :

comment accéder à la propriété `style` de notre élément ?

Eh bien de la même manière que pour accéder à n'importe quelle propriété de notre élément :

Code : JavaScript

```
element.style; // On accède à la propriété « style » de l'élément «element »
```

Une fois que l'on a accédé à notre propriété, comment modifier les styles CSS ? Eh bien tout simplement en écrivant leur nom et en leur attribuant une valeur, `width` (pour la largeur) par exemple :

Code : JavaScript

```
element.style.width = '150px'; // On modifie la largeur de notre élément à 150px
```

Sauvegarder ce fichier sous le nom « util.js » dans le même dossier que « exemple.html » (qui est dans « TP1 »)

Tester la page « exemple.html » sous Firefox, cliquer sur le 1er bouton : est-ce que ça marche ? NON !
Faire le nécessaire pour faire fonctionner ce bouton.

IMPORTANT : on rappelle que pour lier un fichier « xyz.js » à un document html, il faut ajouter ceci dans le fichier HTML : `<script src='xyz.js'></script>`

Développer et tester une fonction pour chaque bouton dont voici les détails :

- Fond de page : gris : changer le fond de la page (body) et mettre la couleur gris (#ccc)

- Combien il y a de paragraphes : afficher le nombre des balises <p> contenu dans ce document
 - Couleur de fond des parag : jaune ? : Modifier la couleur de fond (background-color) de chaque paragraphe (donc boucle) et mettre 'yellow' comme couleur
 - Couleur de fond des parag pairs : rouge ? : Même principe, mais à appliquer sur les paragraphes pairs
 - Cacher les parag impairs : Changer le style des paragraphes impairs en mettant « display » à 'none'
 - Ajouter un lien hypertexte : Permet d'ajouter à la suite des paragraphes une balise « a » possédant Les attributs suivants (id='jQuery_link' , href = ' https://jquery.com/ ', title = ' Découvrez le Site de jQuery ! ')
- et contenant un texte de lien «La référence de jQuery sur le web».

Indication :

Avec le DOM, l'ajout d'un élément HTML se fait en trois temps :

1. On crée l'élément ; // document.createElement(____)
2. On lui affecte des attributs ; //
3. On l'insère dans le document; // ____.appendChild(____)

pour ajouter du texte dans le DOM se servir de la fonction suivante :

- Go : permet de définir une couleur de fond jaune à chaque occurrence du mot (saisi dans le champ) dans le contenu de la page. Pour commencer créer une nouvelle classe de style appelé « .selection » et y définir le jaune comme couleur de fond.

Code CSS :

```
<style type="text/css">
.selection
{
background-color:yellow;
font-family: arial, verdana, sans-serif;
font-style:italic;
}
</style>
```

Solution

Voici donc un pseudo-algorithme permettant d'automatiser ce processus :

- Récupérer le mot à chercher
- Préparer la nouvelle version du mot (en l'entourant avec un span et en lui appliquant la classe)
- Récupérer tous les paragraphes de la page
 - Pour chaque paragraphe faire
 - Récupérer le contenu HTML du paragraphe
 - Remplacer (dans ce contenu) chaque itération du mot par sa nouvelle version
 - Remettre le nouveau code HTML

FinPour

Fonctions Javascript à utiliser :

- **e.innerHTML** : retourne ou modifie le contenu HTML d'un élément (balise)
- **s.replace("aaa", "bbb")** : crée une copie de la chaîne s, remplace la chaîne "aaa" par "bbb" (qui peuvent être des variables) et retourne une nouvelle chaîne.

Tester la page.

Améliorations

On remarque que seule la 1ère occurrence du mot est trouvée, c'est le principe de fonctionnement de la fonction replace() ! Voilà comment résoudre le problème :

Syntaxe

```
reg=new RegExp(String motif[, String type])
```

```
res = s.replace(reg, mot2);
```

Description

Objet Expression régulière

Le paramètre motif décrit le format de chaîne à trouver.

Le paramètre type décrit le type d'expression régulière.

Si type vaut "g", l'expression sera analysée globalement sur l'ensemble de la chaîne.

Si type vaut "i", l'expression sera analysée indifféremment sur les majuscules ou les minuscules.

type peut donc valoir "", "g", "i" ou "gi"